35th Annual International Conference of the IEEE EMBS
Osaka, Japan, 3 - 7 July, 2013

Investigating the role of firing-rate normalization and dimensionality reduction in brain-machine interface robustness

Jonathan C. Kao[1], *Student Member, IEEE*, Paul Nuyujukian[2,3], *Member, IEEE*, Sergey Stavisky[4], Stephen I. Ryu[1,7], Surya Ganguli[1,4,5,6] & Krishna V. Shenoy[1,2,4,6], *Senior Member, IEEE*

***Abstract*— The intraday robustness of brain-machine interfaces (BMIs) is important to their clinical viability. In particular, BMIs must be robust to intraday perturbations in neuron firing rates, which may arise from several factors including recording loss and external noise. Using a state-of-the-art decode algorithm, the Recalibrated Feedback Intention Trained Kalman filter (ReFIT-KF) [1] we introduce two novel modifications: (1) a normalization of the firing rates, and (2) a reduction of the dimensionality of the data via principal component analysis (PCA). We demonstrate in online studies that a ReFIT-KF equipped with normalization and PCA (NPC-ReFIT-KF) (1) achieves comparable performance to a standard ReFIT-KF when at least 60% of the neural variance is captured, and (2) is more robust to the undetected loss of channels. We present intuition as to how both modifications may increase the robustness of BMIs, and investigate the contribution of each modification to robustness. These advances, which lead to a decoder achieving state-of-the-art performance with improved robustness, are important for the clinical viability of BMI systems.**

I. Introduction

Brain-machine interfaces (BMIs) map cortical activity into meaningful control signals, via a decoder, to drive computer cursors or robotic limbs [1]–[7]. Significant translational work has helped to improve the performance of BMIs, which is important to their clinical viability [1], [3], [7]. However, intraday and interday stability of BMIs remain important hurdles to address, since the potential for instability in neuron firing rates, or the loss of neurons, can cripple a high-performing decoder [8], [9]. In this study, we address undetected intraday neuron loss: after a decoder is trained at the beginning of the day, we would like it to be robust to the unexpected loss of neurons without having to be retrained. Such neuron loss could occur, e.g. from a change in spike amplitude, rendering the spike event no-longer detected (as in spike sorting or threshold crossings with static parameters) [10]. One might employ an additional algorithm to detect the loss of a neuron; here, however, we address the case where a decoder is more robust to neuron loss without any additional algorithms. We further recognize that algorithms to detect neuron recording loss may serve as an additional source of robustness. Nevertheless, the goal of this study is to present a decoder which achieves state-of-the-art performance while being more robust than current state-of-the-art techniques to undetected channel loss.

[1]Dept. of Electrical Engineering, [2]Dept. of Bioengineering, [3]School of Medicine, [4]Neurosciences Prog., [5]Dept. of Applied Physics, [6]Dept. of Neurobiology, Stanford University, Stanford, CA; [7]Dept. of Neurosurgery, Palo Alto Medical Foundation, Palo Alto, CA

This work was supported by a National Science Foundation (NSF) Graduate Research Fellowship (J.C.K), Stanford Medical Scientist Training Program (P.N.), NSF Integrative Graduate Education and Research Trainee Program (S.S), NIH CRCNS R01-NS054283, NIH Pioneer Award 1DP1OD006409, DARPA REPAIR contract N66001-10-C-2010 (K.V.S).

II. Methods

A. Online neural recordings

All procedures and experiments were approved by the Stanford University Institutional Animal Care and Use Committee. Experiments were conducted with an adult male rhesus macaque (J), implanted with two 96 electrode Utah arrays (Blackrock Microsystems Inc., Salt Lake City, UT) using standard neurosurgical techniques [4]. Monkey J was implanted $26-33$ months prior to the experiments. One electrode array was implanted in PMd and the other in M1, as estimated visually from local anatomical landmarks.

Monkey J was trained to make point-to-point reaches in a 2D plane with a virtual cursor controlled by the contralateral arm or by a neural decoder [11]. The virtual cursor and targets were presented in a 3D environment (MSMS, MDDF, USC, Los Angeles, CA) described in [1]. Hand position data were measured with an infrared reflective bead tracking system (Polaris, Northern Digital, Ontario, Canada). Behavioral control and neural decode were run on separate PCs using the Simulink/xPC platform (Mathworks, Natick, MA). Neural data were initially processed by the Cerebus recording system (Blackrock Microsystems Inc., Salt Lake City, UT) according to specifications described in [1]. Spike events were detected by setting a threshold value to $-4.5\times$ the RMS voltage of the channel. The number of spike events were counted in non-overlapping 5 ms bins.

B. Decoder task

All BMI experiments reported in this paper utilize the Recalibrated Feedback Intention Trained Kalman filter (ReFIT-KF) algorithm [1] which utilizes intention estimation techniques and a control-feedback approach to increase the performance of Kalman filters (KFs). While training sets were collected using a center-out-and-back task with 8 targets, all performance sessions reported in this paper utilized a cued target selection task with a 5×5 grid of targets, one of which was the correct target on any given trial. The acceptance windows (one for each target) were square boxes with 4.8 cm

978-1-4577-0216-7/13/$26.00 ©2013 IEEE

sides, and were tiled contiguously in the workspace with no overlap (so that the workspace is 24 cm $\times$ 24 cm). At any point in time, any target could be selected by dwelling on the target for 450 ms, unless the cursor was outside of the workspace (i.e. not on any target). There was a "grace time" of 200 ms at the start of all trials during which dwell-time was not counted, to account for reaction time. Targets were presented for a maximum of 5 s before a new target was shown. Because any target can be selected during the task, each selection conveys $\log_2(25)$ bits of information, enabling a bitrate calculation for the task. We conservatively assumed that any incorrect selection must be compensated by a correct selection (as when selecting the backspace key to undo an incorrect keystroke on a keyboard). Given that T is the duration of a sequence in which r correct targets were selected and s incorrect targets were selected, the bitrate is $I = \frac{\log_2(25)(r-s)}{T}$ as long as $r > s$, and $I = 0$ otherwise. By this performance metric, a success rate of 50% on the task corresponds to $I = 0$.

To simulate channel loss, we randomly chose channels and set their firing rates to zero, as if they were no longer firing. The decoder was not retrained, as we were testing for robustness in the case of undetected channel loss.

C. Normalization of channel firing rates

Let $y_k \in \mathbb{R}^N$ be the number of observed spike events in time bin k for N channels. To perform firing rate normalization, we first averaged the neural data in each trial over 16 conditions, corresponding to the center-out-and-back task with 8 targets. We then performed a normalization on each channel's firing rate [12], $(y_k)_i$ (i denoting the i^{th} element of vector y_k, for $i = 1, \ldots, N$), given by

$$(z_k)_i = \frac{(y_k)_i}{r_i + \nu} \tag{1}$$

where r_i is the maximum observed firing rate minus the minimum observed firing rate in the condition-averaged neural data for channel i, and ν is a non-negative scalar.

D. Dimensionality reduction via principal component analysis

To reduce the dimensionality of the neural data, we performed principal component analysis on the normalized condition-averaged trajectories. The normalization was used so that channels with higher firing rates, which tend to have higher variance, do not dominate the covariance of the data [13]. Further, PCA was performed on the condition-averaged trajectories so that more emphasis would be placed on the variance between task conditions rather than single-trial variance. Let (μ, Σ) denote the mean and the covariance matrix of the normalized condition-averaged trajectories, and let the eigenvalue decomposition of the covariance matrix be $\Sigma = U \Lambda U^T$. $\Lambda \in \mathbb{R}^{N \times N}$ is diagonal populated with the eigenvalues $\Lambda_{ii} = \lambda_i$ corresponding to the amount of variance captured in dimension i and are sorted so that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$. The columns of $U \in \mathbb{R}^{N \times N}$ are called the PC vectors, the first D of which are stored in $P \in \mathbb{R}^{N \times D}$. To compute a low-dimensional projection of the binned spike-counts in $\mathbb{R}^D$, we calculated $s_k = P^T(z_k - \mu)$. We then used the s_k's as the observation of a ReFIT-KF (as opposed to binned spike counts, y_k); for convenience, we call this decoder the NPC-ReFIT-KF. We refer to $s_k \in \mathbb{R}^D$ as the *neural state*, since it captures something meaningful about how the activity of a network of spiking neurons evolves.

III. Intuitions for NPC-ReFIT-KF robustness

It is important that the NPC-ReFIT-KF, as a decoder which excludes neural variance by virtue of reducing the dimensionality of the neural data, be of comparable performance to the ReFIT-KF. Previous studies suggest that the dimensionality of PMd and M1 during simple reaching tasks is approximately $D = 10-20$ [13]. Hence, although reducing the dimensionality of the data from N to D results in the exclusion of neural variance, much of the meaningful signal variance is maintained in the top $10-20$ dimensions. Therefore, we predict that the performance of the D-dimensional NPC-ReFIT-KF will be similar to that of the ReFIT-KF, until the point where D is low enough such that meaningful signal variance is lost. For this work, we use $D = 20$, capturing approximately 60% of the neural variance.

A. The effect of firing-rate normalization on decoder weights

Normalization by $r_i + \nu$ (Eqn. 1) can be viewed as a regularization of the firing rates. For channels with strong responses, the normalization maps their firing rate to approximately the unity interval, so that strongly firing channels have similar dynamic range. (Alternatively, this can be viewed as very approximate variance normalization.) On the other hand, the parameter ν (chosen to be $\nu = 20$ spikes/s) causes the activity of weakly firing channels to be mapped to a smaller than unity dynamic range of firing. Therefore, the normalization penalizes weakly firing channels, as well as channels with relatively high variance but little modulation. This is an important pre-processing step to principal component analysis, so that high variance channels do not dominate the dimensionality reduction [13].

The normalization, which causes strongly firing channels to have similar dynamic range, also empirically decreases the variance in channel contribution to the decoder (i.e. the spread in channel contribution across all channels) by approximately 15%. An intuition for how this may affect robustness is as follows: when a small set of channels are assigned relatively strong weights in the decoder relative to all other channels, the loss of such strong weighted channels may cripple decoder performance. On the other hand, when the decoder weights for informative channels become more similar, the robustness of the decoder may improve. However, to ultimately assess how the normalization of firing rates affects robustness to channel loss, one should analyze how driving the decoder with the perturbed normalized firing rates affects the decoded velocity. In §III.C, we compare how the ReFIT-KF and the NPC-ReFIT-KF (which incorporates normalization) project perturbed neural data to velocity errors.

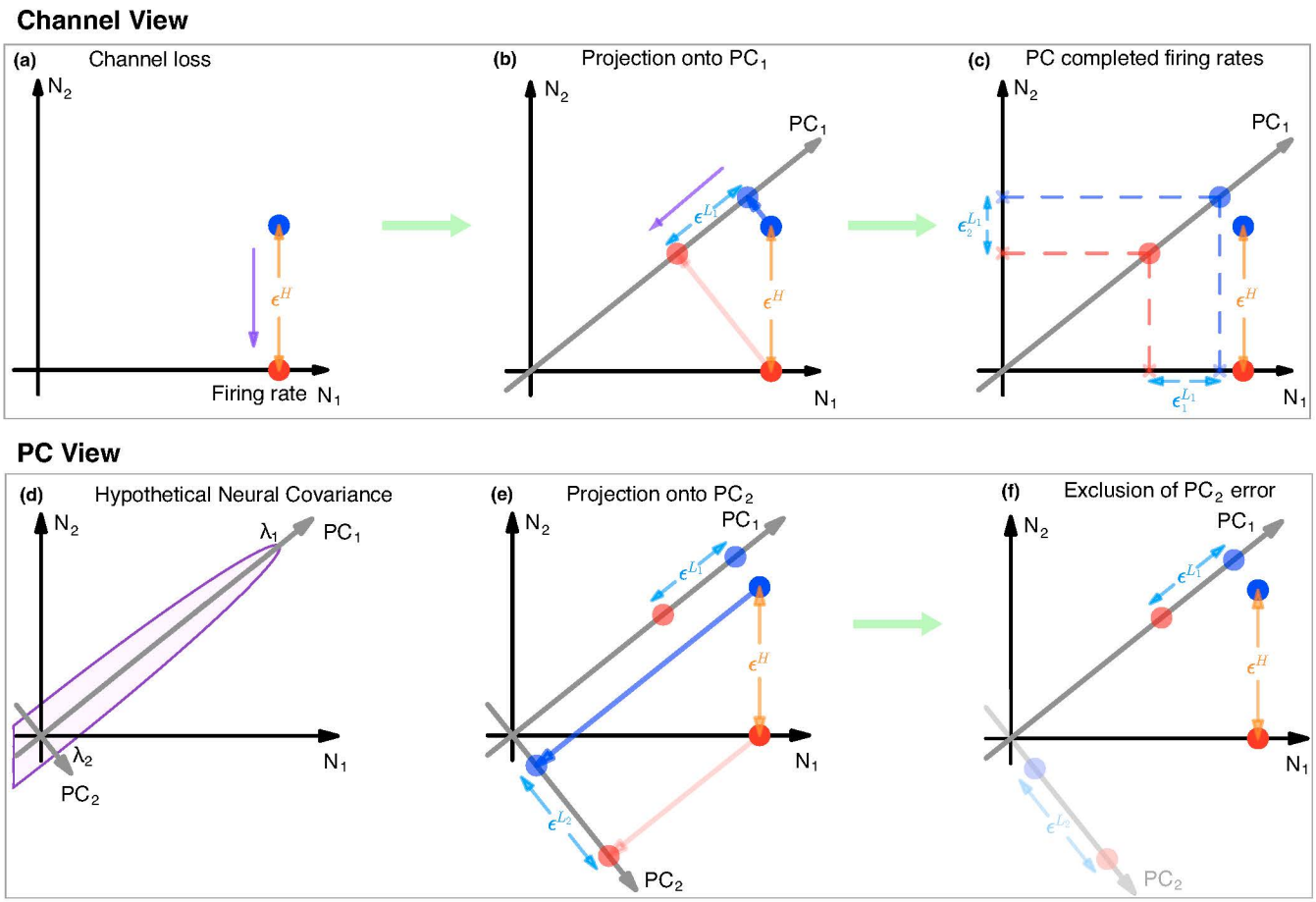

Fig. 1. **Two views of the potential benefit of PCA**: (a) The x-axis denotes the firing rate of channel N_1, and the y-axis that of channel N_2. The blue dot represents a firing rate observation, while the red dot represents the firing rate observation in the case when channel N_2 is lost (so that $N_2 = 0$). Losing N_2 introduces error ϵ^H in the N_2 direction. (b) Both the unperturbed observation (blue) and the perturbed observation (red) are projected onto PC_1. The orientation of PC_1 indicates that N_1 and N_2 tend to fire together. (c) By virtue of the projection, a non-zero firing rate is inferred for N_2 in the case of channel loss (red), reducing the error in the observation of N_2, i.e. $\|\epsilon_2^{L_1}\|_1 < \|\epsilon^H\|_1$. However, there is an error introduced in the observation of N_1, i.e. $\|\epsilon_1^{L_1}\|_1 > 0$. Hence, large errors in any single channel become distributed, in a mitigated fashion, amongst the other channels. (d) For the same scenario, we show the (hypothetical) covariance ellipsoid, indicating that the neural state tends to vary along PC_1. (e) We show the projection of ϵ^H onto PC_2, which corresponds to introducing an error ϵ^{L_2} in the direction of PC_2. We note that $\|\epsilon^H\|_2^2 = \|\epsilon^{L_1}\|_2^2 + \|\epsilon_2^{L_2}\|_2^2$. (f) However, if the decoder only uses PC_1, the error introduced into PC_2 does not perturb the decoder. From here, it is apparent that $\|\epsilon^{L_1}\|_2 \leq \|\epsilon^H\|_2$.

B. Dimensionality reduction incorporates unsupervised network-level information into the BMI

To be more clear in explaining intuition as to how dimensionality reduction via PCA may increase robustness, we ignore normalization and mean-subtraction for this section only. Hence, we let $s_k = P^T y_k$. It is straightforward to account for normalization and mean-subtraction in the analyses of this section.

1) Equivalent NPC-ReFIT-KF: In the case of undetected channel loss, a channel which previously reported a non-zero firing rate will report a zero firing rate, causing the performance of a BMI to decline. To analyze this scenario, we consider the steady-state form of the ReFIT-KF which converges on the order of seconds [14]. In steady-state, the KF is a linear feedback filter of the form

$$x_k = M_1^H x_{k-1} + M_2^H y_k \tag{2}$$

Therefore, we may write the NPC-ReFIT-KF as:

$$\begin{aligned} x_k &= M_1^L x_{k-1} + M_2^L s_k && (3)\\ &= M_1^L x_{k-1} + M_2^L P^T y_k && (4)\\ &= M_1^L x_{k-1} + \widetilde{M}_2^L y_k^{PC} && (5) \end{aligned}$$

with $\widetilde{M}_2^L = M_2^L P^T$ and $y_k^{PC} = PP^T y_k$, since $P^T P = I$ by the orthonormality of the P_i's. Hence, the low-dimensional decoder can be viewed as operating on projected firing rates y_k^{PC}. (Note that by the presence of P^T in $\widetilde{M}_2^L$, which causes the neural data to project to kinematics through the PCs, we have that $\widetilde{M}_2^L y_k^{PC} = \widetilde{M}_2^L y_k$. Analyzing $y_k^{PC} \in \text{range}(P)$ is helpful in understanding the projection.)

2) PCA performs a firing rate completion in the channel space: The term y_k^{PC} is significant because it incorporates network-level information, P, into the firing rates. PP^T is an orthogonal projector onto range(P), which passes through the origin. When a channel is erroneously reporting a firing

rate of 0 (as in undetected channel loss), the projection onto range(P) infers a firing rate for this channel by projecting *all* of the neural data into the underlying subspace in which the neural state evolves. This is because the coefficients of P_i capture broad network-level correlations, e.g. $\frac{P_{ij}}{P_{ik}}$ can be viewed as a typical ratio of how neuron j and k fire together along dimension i. Intuitively, when 25% of the channels are lost, knowing how the remaining 75% of the channels fire reveal, through P, how the remaining 25% of channels may have fired, a process we call "PC completion." A hypothetical example of PC completion is illustrated in Fig. 1a-c for the case of $N = 2$ and $D = 1$. In our case, with $N = 192$ and $D = 20$, there are $(N - 1) \times D$ coefficients which reveal a network-based relationship between the firing rate of one channel and the firing rates of the other 191 channels.

It is important to note that the effect of the projection is reciprocal. That is, while unperturbed channels can mitigate the error in a channel reporting 0 firing rate, similarly, the channel reporting a 0 firing rate, through P, will introduce a small error in the estimation of the firing rate of the unperturbed channels, as in Fig. 1c. Hence, for a reasonable PC completion, a significant proportion of unperturbed channels should be available, so that there are sufficient observations of true underlying firing rates to mitigate errors introduced by the undetected errors in perturbed channels.

3) PC view – the low-dimensional decoder ignores error along higher-order PC dimensions: The low-dimensional projection extracts much of the meaningful signal variance in the top D dimensions, provided D is sufficiently large, and ignores the neural variance in the remaining $N - D$ dimensions. Hence, a potential benefit of low-dimensional decoding is that perturbations which have large projections to unused PCs do not affect the decoder. We illustrate this in Fig. 1d-f, where the low-dimensional neural state tends to move along PC_1 (i.e. $\lambda_1 \gg \lambda_2$). Then the decoder, which only considers the top D dimensions (e.g., PC_1 in Fig. 1d-f), does not take into account the neural perturbation along the remaining $N - D$ dimensions (e.g. ignoring PC_2 in Fig. 1d-f). We note that, in Fig. 1d-f, ε^{L_2} and ε^{L_1} represent the decomposition of ε^H along the PCs. Hence, a portion of the error perturbation introduced by channel loss ε^H is attenuated by virtue of ignoring the projection of the error onto higher dimensional PCs. Intuitively, we have that $\|P^T \varepsilon^H\|_2 \leq \|\varepsilon^H\|_2$.

C. The NPC-ReFIT-KF mitigates velocity perturbation introduced by channel loss

We must ultimately understand how the channel loss perturbation is projected to the decoded velocity. For velocity-based KFs, such as ReFIT-KF and NPC-ReFIT-KF, M_2^H and M_2^L (as in Eqns. (2) and (3) respectively) map observations (y_k and s_k) to cursor velocity only. To analyze how channel loss perturbs velocity, we define $\varepsilon_{y_k} = y_k^0 - y_k$, where y_k are the firing rates if there is no channel loss, and y_k^0 are the firing rates with channel loss. Then, ε_{y_k} is the firing rate perturbation (error) introduced by channel loss.

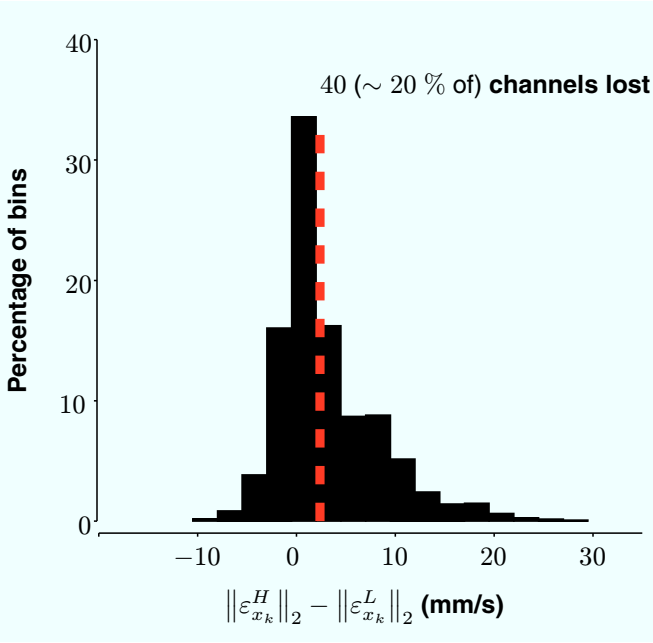

Fig. 2. **Firing rate perturbation comparison** For 40 (∼ 20 % of) channels lost, we project the firing rate errors through both the NPC-ReFIT-KF and ReFIT-KF to see how they map to velocity errors, $\|\varepsilon^L_{x_k}\|_2$ and $\|\varepsilon^H_{x_k}\|_2$, respectively. A histogram of $\|\varepsilon^H_{x_k}\|_2 - \|\varepsilon^L_{x_k}\|_2$ over time bins k is shown to indicate that the NPC-ReFIT-KF mitigates error in cursor speed introduced by channel loss by more than 5 mm/s over the ReFIT-KF approximately 30% of the time, and by more than 10 mm/s over 10% of the time. The mean difference, 2.4 mm/s (red dashed line) is significantly different than 0 by a t-test ($p < 0.05$).

For the ReFIT-KF decoder, the perturbation to velocity as a result of channel loss for time bin k is $\varepsilon^H_{x_k} = M_2^H \varepsilon_{y_k}$. For the NPC-ReFIT-KF, the same quantity is $\varepsilon^L_{x_k} = M_2^L P^T R \varepsilon_{y_k}$ (where R a diagonal matrix with $R_{ii} = \frac{1}{r_i + \nu}$, i.e. the firing-rate normalization for channel i). For 40 channels lost (approximately 20 % of channels lost), we show in Fig. 2 the difference in the magnitude of the decoded velocity error resulting from channel loss between the ReFIT-KF and NPC-ReFIT-KF decoders. We see that the error in cursor speed resulting from channel loss is significantly greater under the ReFIT-KF when compared to the error in cursor speed resulting from the NPC-ReFIT-KF, i.e. $\left\langle \|\varepsilon^H_{x_k}\|_2 \right\rangle > \left\langle \|\varepsilon^L_{x_k}\|_2 \right\rangle$ where $\langle \cdot \rangle$ denotes averaging over time bins k.

IV. Results

A. NPC-ReFIT-KF achieves state-of-the-art performance while capturing approximately 60 % of the neural variance

Without channel loss, the performance of ReFIT-KF and NPC-ReFIT-KF is comparable. We noticed no significant difference in performance when using $D = 20$ PCs (capturing approximately 60 % of the variance) as shown in Fig. 3a (for 100% of the neurons). To calibrate the reader, an information rate of 4 bps on this task corresponds to approximately one effective correct target selection every ~ 1.2 s.

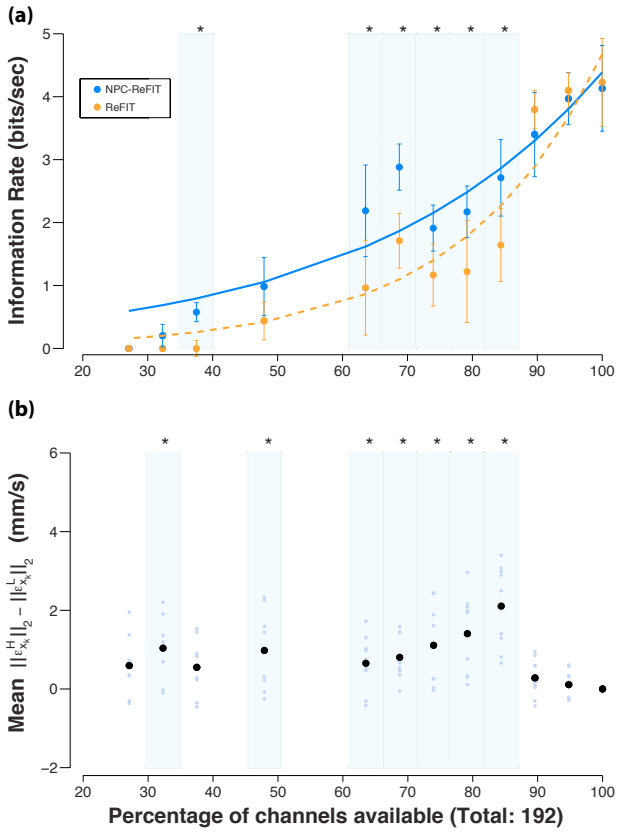

Fig. 3. **Performance under channel loss.** (a) The bitrate for both the ReFIT-KF (orange) and the NPC-ReFIT-KF (blue) under channel loss. The NPC-ReFIT-KF sustains higher bitrates in the face of channel loss. A * indicates significance by a t-test ($p < 0.05$). (b) The corresponding per-block average difference in speed error resulting from channel loss. Though the speed differences are on average < 1 cm/s, Fig. 2 shows that the ReFIT-KF speed error is often 1 cm/s greater than that of the NPC-ReFIT-KF during the trial. When $60 - 85\%$ of the channels are available, a significance in the difference in mean speed errors corresponds to a significant performance difference on the bitrate task.

B. The NPC-ReFIT-KF is more robust to undetected channel loss than the ReFIT-KF

In Fig. 3a we show the bitrate for both the ReFIT-KF and the NPC-ReFIT-KF under channel loss for a single randomized channel loss order with repeated measurements. We see that for this randomized dropping order, the NPC-ReFIT-KF is in general more robust channel loss. Correspondingly in Fig. 3b, the difference between the mean cursor speed error as a result of channel loss, $\left\langle \left\| \varepsilon_{x_k}^H \right\|_2 \right\rangle - \left\langle \left\| \varepsilon_{x_k}^L \right\|_2 \right\rangle$ for each session is shown. When $60 - 85\%$ of the channels are available, a significance in the mean cursor speed error between ReFIT-KF and NPC-ReFIT-KF (Fig. 3b) corresponds to a significant performance difference (Fig. 3a).

To show that in general, the NPC-ReFIT-KF is more robust than the ReFIT-KF when random channels are dropped, we randomized 14 different dropping orders of 40 ($\sim$ 20 % of) channels and measured a bitrate from 200 trials for both decoders, as shown in Fig. 4. The NPC-ReFIT-KF decoder generally outperforms the ReFIT-KF (mean difference: 0.74 bps, significant at $p < 0.05$).

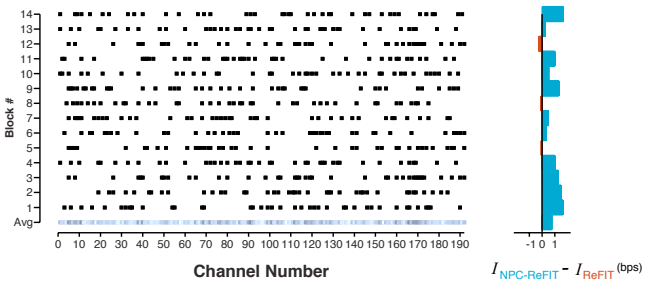

Fig. 4. **Performance of ReFIT-KF vs NPC-ReFIT-KF for randomly losing** 40 ($\sim$ 20 % **of) channels.** Each row indicates a different experimental block of 400 trials, during which the channel number with a black dot denotes that this channel was lost (i.e. its firing rate was set to zero). The difference in bitrate performance for each block is indicated by the bar plot. The NPC-ReFIT-KF achieves significantly higher mean bitrate than the ReFIT-KF across the aggregate of drop orders.

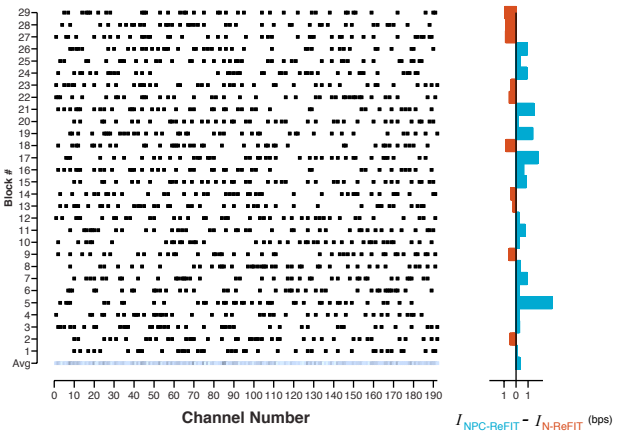

Fig. 5. **Performance of ReFIT-KF with normalized firing rates vs NPC-ReFIT-KF for randomly losing** 40 ($\sim$ 20 % **of) channels.** As in Fig. 4, each row signifies a bitrate measurement for two decoders with 40 channels lost. Averaging across all random dropping orders, NPC-ReFIT-KF achieves a higher mean bitrate than ReFIT-KF, however, the difference is not significant.

C. ReFIT-KF with normalized firing rates vs NPC-ReFIT-KF

To further investigate if the robustness is primarily a result of the normalization or the dimensionality reduction, we compared the bitrate of a ReFIT-KF whose observations are $z_k - \mu$ (see Eqn. 1) with the NPC-ReFIT-KF, whose observations are $s_k = P^T(z_k - \mu)$. Hence, the two decoders only differ in that the NPC-ReFIT-KF utilizes dimensionality reduction through P. We evaluated the bitrate for 29 different dropping orders of 40 ($\sim$ 20 % of) channels as shown in Fig. 5. We found that the NPC-ReFIT-KF on average achieved a higher average bitrate (0.3 bps) which was not significant ($p \approx 0.09$), indicating that dimensionality reduction at $D = 20$ did not significantly improve the robustness of the decoder.

V. Conclusion and Future Work

We show that a NPC-ReFIT-KF, which utilizes firing normalization and dimensionality reduction on binned spike

counts, performs comparably to a ReFIT-KF in a grid selection task. We further show that such a decoder is in general more robust to the undetected loss of channels.

A natural question is to further investigate how dimensionality reduction contributes to robustness. For example, the robustness may be significantly affected by changing the dimensionality D, since increasing D increases the number of coefficients for PC-completion. Future work will also be aimed at more deeply understanding the relative contribution of normalization and dimensionality reduction to robustness. Finally, future work should also address how these effects vary across subjects with differing ensemble neural correlations, vary with task complexity, and vary with differing types of decoders (for example, nonlinear decoders [15]).

ACKNOWLEDGEMENT

We thank M. Mazariegos, J. Aguayo, C. Sherman & E. Morgan for expert surgical assistance & veterinary care, B. Oskotsky for IT support, and B. Davis, E. Casteneda & S. Eisensee for administrative assistance.

REFERENCES

[1] V. Gilja, P. Nuyujukian, C. A. Chestek, J. P. Cunningham, B. M. Yu, J. M. Fan, M. M. Churchland, M. T. Kaufman, J. C. Kao, S. I. Ryu, and K. V. Shenoy, "A high-performance neural prosthesis enabled by control algorithm design," *Nature Neuroscience*, vol. 15, pp. 7–10, Nov. 2012.

[2] J. M. Carmena, M. A. Lebedev, R. E. Crist, J. E. O'Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. Henriquez, and M. A. L. Nicolelis, "Learning to control a brain-machine interface for reaching and grasping by primates," *PLoS biology*, vol. 1, p. E42, Nov. 2003.

[3] L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. A. Mukand, M. Saleh, A. H. Caplan, A. Branner, D. Chen, R. D. Penn, and J. P. Donoghue, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia," *Nature*, vol. 442, pp. 164–71, July 2006.

[4] G. Santhanam, S. I. Ryu, B. M. Yu, A. Afshar, and K. V. Shenoy, "A high-performance brain-computer interface," *Nature*, vol. 442, pp. 195–198, July 2006.

[5] M. Velliste, S. Perel, M. C. Spalding, A. S. Whitford, and A. B. Schwartz, "Cortical control of a prosthetic arm for self-feeding," *Nature*, vol. 453, pp. 1098–101, June 2008.

[6] J. E. O'Doherty, M. A. Lebedev, P. J. Ifft, K. Z. Zhuang, S. Shokur, H. Bleuler, and M. A. L. Nicolelis, "Active tactile exploration using a brain-machine-brain interface," *Nature*, vol. 479, pp. 228–31, Nov. 2011.

[7] L. R. Hochberg, D. Bacher, B. Jarosiewicz, N. Y. Masse, J. D. Simeral, J. Vogel, S. Haddadin, J. Liu, S. S. Cash, P. van der Smagt, and J. P. Donoghue, "Reach and grasp by people with tetraplegia using a neurally controlled robotic arm," *Nature*, vol. 485, pp. 372–5, May 2012.

[8] A. M. Green and J. F. Kalaska, "Learning to move machines with the mind," *Trends in Neurosciences*, vol. 34(2), pp. 61–75, 2011.

[9] M. A. L. Nicolelis and M. A. Lebedev, "Principles of neural ensemble physiology underlying the operation of brain-machine interfaces," *Nature reviews. Neuroscience*, vol. 10, pp. 530–40, July 2009.

[10] C. A. Chestek, V. Gilja, P. Nuyujukian, J. D. Foster, J. M. Fan, M. T. Kaufman, M. M. Churchland, Z. Rivera-Alvidrez, J. P. Cunningham, S. I. Ryu, and K. V. Shenoy, "Long-term stability of neural prosthetic control signals from silicon cortical arrays in rhesus macaque motor cortex," *Journal of neural engineering*, vol. 8, p. 045005, Aug. 2011.

[11] J. P. Cunningham, P. Nuyujukian, V. Gilja, C. A. Chestek, S. I. Ryu, and K. V. Shenoy, "A closed-loop human simulator for investigating the role of feedback control in brain-machine interfaces," *Journal of Neurophysiology*, vol. 105, pp. 1932–1949, 2011.

[12] M. M. Churchland, J. P. Cunningham, M. T. Kaufman, J. D. Foster, P. Nuyujukian, S. I. Ryu, and K. V. Shenoy, "Neural population dynamics during reaching," *Nature*, vol. 487, pp. 51–6, July 2012.

[13] B. M. Yu, J. P. Cunningham, G. Santhanam, S. I. Ryu, and K. V. Shenoy, "Gaussian-prpocess factor analysis for low-dimensional single-trial analysis of neural population activity," *Journal of Neurophysiology*, vol. 102, pp. 612–635, 2009.

[14] W. Q. Malik, W. Truccolo, E. N. Brown, and L. R. Hochberg, "Efficient decoding with steady-state Kalman filter in neural interface systems," *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 19, pp. 25–34, Feb. 2011.

[15] D. Sussillo, P. Nuyujukian, J. M. Fan, J. C. Kao, S. D. Stavisky, S. Ryu, and K. Shenoy, "A recurrent neural network for closed-loop intracortical brain-machine interface decoders," *Journal of neural engineering*, vol. 9, p. 026027, Apr. 2012.